Data Structures

Lists

Question Number 1

In [32]:

```
# Creating an empty list
empty_list = list()

print("Empty List: ",empty_list)
```

```
Empty List:  []
```

Question Number 2

In [37]:

```
# Creating a list
my_list = [1, 2, 3]

# Adding an element to the end of the list
my_list.append(4)

print(my_list)
```

```
[1, 2, 3, 4]
```

Question Number 3

In [40]:

```
# Creating a list
my_list = [1, 2, 4, 5]

# Inserting an element at index 2
my_list.insert(2, 3)

print(my_list)
```

```
[1, 2, 3, 4, 5]
```

Question Number 4

In [47]:

```
# Creating a list
my_list = [1, 2, 3, 4, 3, 5]

# Removing the element with value 3
my_list.remove(3)

print(my_list)
```

```
[1, 2, 4, 3, 5]
```

Question Number 5

In [51]:

```
# Creating a list
my_list = [1, 2, 3, 4, 5]
```

```
# Removing the element at index 2
removed_element = my_list.pop(2)

print(my_list)
print(removed_element)
```

```
[1, 2, 4, 5]
3
```

Question Number 6

In [54]:

```
# Creating a list
my_list = [10, 20, 30, 40]

# Accessing the first element
first_element = my_list[0]

print(first_element)
```

```
10
```

Question Number 7

In [59]:

```
# Creating a list
my_list = [10, 20, 30, 40]

# Accessing the first element
last_element = my_list[3]

print(last_element)
```

```
40
```

Question Number 8

In [62]:

```
# Creating a list
my_list = [10, 20, 30, 40]

# Accessing the element at specific index
specific_element = my_list[1]

print(specific_element)
```

```
20
```

Question Number 9

In [65]:

```
# Creating a list
my_list = [10, 20, 30, 40, 50]

# Updating the element at index 2
my_list[2] = 35

print(my_list)
```

```
[10, 20, 35, 40, 50]
```

Question Number 10

In [68]:

```
# Creating a list with numbers from 1 to 10
my_list = list(range(1, 11))

print(my_list)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Question Number 11

In [71]:

```
# Creating a list
my_list = [10, 20, 30, 40, 50]

# Finding the length of the list
print(len(my_list))
```

```
5
```

Question Number 12

In [74]:

```
if not my_list:
    print("List is empty")
else:
    print("List is not empty")
```

```
my_list is not empty
```

Question Number 13

In [79]:

```
list2 = [12,3,3,332,3,4,5,5]
element = 5

# Count occurrences of the element
count = list2.count(element)

print(f"The element {element} appears {count} times in the list.")
```

```
The element 5 appears 2 times in the list.
```

Question Number 14

In [92]:

```
list2 = [12, 3, 3, 332, 3, 4, 5, 5]
# Returns a new sorted list
sorted_list = sorted(list2)
print(sorted_list)
```

```
[3, 3, 3, 4, 5, 5, 12, 332]
```

Question Number 15

In [97]:

```
#Reversing the list items
list2.reverse()
list2
```

Out[97]:

```
[12, 3, 3, 332, 3, 4, 5, 5]
```

Question Number 16

In [100…

```
list2 = [12, 3, 3, 332, 3, 4, 5, 5]
#Slicing the list to show first three list items
print(list2[0:3])
```

```
[12, 3, 3]
```

Question Number 17

In [123…

```
list2 = [12, 3, 3, 332, 3, 4, 5, 5]
#Slicing the list to show last three list items
print(list2[-3:])
```

```
[4, 5, 5]
```

Question Number 18

In [126…

```
list1=[1,23,4,3]
list2=[2,3,4,4]
#Concatenating two lists
list3 = list1+list2
print(list3)
```

```
[1, 23, 4, 3, 2, 3, 4, 4]
```

Question Number 19

In [129…

```
list1=[1,23,4,3]
#printing the max list item from list1
print(max(list1))
```

```
23
```

Question Number 20

In [132…

```
list1=[1,23,4,3]
#printing the min list item from list1
print(min(list1))
```

```
1
```

Tuples

Question Number 21

In [135…

```
# Creating an empty tuple
empty_tuple = tuple()

print("Empty Tuple: ",empty_tuple)
```

```
Empty Tuple:  ()
```

Question Number 22

In [141…

```python
# Creating a tuple with multiple elements
my_tuple = (1, 2, 3, "four", 5.0)

print(my_tuple)
```

```
(1, 2, 3, 'four', 5.0)
```

Question Number 23

In [144…

```python
#Accessing the first element of tuple
print(my_tuple[2])
```

```
3
```

Question Number 24

In [151…

```python
# Creating a tuple
my_tuple = (10, 20, 30, 40, 50)

# Accessing the last element
last_element = my_tuple[-1]

print(last_element)
```

```
50
```

Question Number 25

In [154…

```python
# Accessing the specific index in a tuple
specific_element = my_tuple[2]

print(specific_element)
```

```
30
```

Question Number 26

In [157…

```python
#Printing the length of tuple
print(len(my_tuple))
```

```
5
```

Question Number 27

In [160…

```python
tup1=(2,3,4,54)
tup2=(3,4,5,5)
#Concatenating two tuples
tup3=tup1+tup2
print(tup3)
```

```
(2, 3, 4, 54, 3, 4, 5, 5)
```

Question Number 28

In [163…

```
# Creating a list
my_list = [1, 2, 3, 4, 5]

# Converting the list to a tuple
my_tuple = tuple(my_list)

print(my_tuple)
```

```
(1, 2, 3, 4, 5)
```

Question Number 29

In [168…

```
# Creating a list
my_tuple = (1, 2, 3, 4, 5)

# Converting the tuple to a list
my_list = list(my_tuple)

print(my_list)
```

```
[1, 2, 3, 4, 5]
```

Question Number 30

In [171…

```
# Creating a tuple
my_tuple = (1, 2, 3, 4, 5)

# Checking if an element exists in the tuple
element_to_check = 3

if element_to_check in my_tuple:
    print(f"{element_to_check} exists in the tuple.")
else:
    print(f"{element_to_check} does not exist in the tuple.")
```

```
3 exists in the tuple.
```

Question Number 31

In [176…

```
# Creating a tuple
my_tuple = (10, 20, 30, 40, 50)

# Finding the index of an element
element_to_find = 30

try:
    index = my_tuple.index(element_to_find)
    print(f"The index of {element_to_find} is {index}.")
except ValueError:
    print(f"{element_to_find} is not in the tuple.")
```

```
The index of 30 is 2.
```

Question Number 32

In [183…

```
tuple2 = (12,3,3,332,3,4,5,5)
element = 5

# Count occurrences of the element
count = tuple2.count(element)

print(f"The element {element} appears {count} times in the tuple.")
```

```
The element 5 appears 2 times in the tuple.
```

Question Number 33

In [189…

```
# Creating a single element tuple
tuple1=(1,)
print(tuple1)
```

```
(1,)
```

Question Number 34

In [191…

```
tuple1=(1,2,3)
tuple2=(1,2,3)
tuple3=(1,2,3)
# Creating a nested tuple
nested_tuple=(tuple1,tuple2,tuple3)
print(nested_tuple)
```

```
((1, 2, 3), (1, 2, 3), (1, 2, 3))
```

Question Number 35

In [194…

```
# Creating a tuple
my_tuple = (10, 20, 30, 40, 50)

# Iterating over elements in the tuple
for element in my_tuple:
    print(element)
```

```
10
20
30
40
50
```

Question Number 36

In [197…

```
# Creating a tuple
my_tuple = (10, 20, 30)

# Unpacking the tuple into individual variables
a, b, c = my_tuple

print(a)
print(b)
print(c)
```

```
10
20
30
```

Question Number 37

In [202…

```
#Reversing a tuple
tuple(reversed(my_tuple))
```

Out[202…

```
(30, 20, 10)
```

Question Number 38

In [213…

```
#Slicing first two elements of tuple
tuple2=(3,4,5)
print(tuple2[0:2])
```

```
(3, 4)
```

Question Number 39

In [218…

```
#Slicing last two elements of tuple
tuple2=(3,4,5)
print(tuple2[-2:])
```

```
(4, 5)
```

Question Number 40

In [221…

```
# Creating a string
my_string = "Hello"

# Creating a tuple from the string
my_tuple = tuple(my_string)

print(my_tuple)
```

```
('H', 'e', 'l', 'l', 'o')
```

Question Number 41

In [224…

```
# Creating an empty dictionary using the dict() function
my_dict = dict()

print(my_dict)
```

```
{}
```

Question Number 42

In [227…

```
dict1={"e":2,"h":4,"hey":"why"}
# Printing the keys of dictionary
print(dict1.keys())
# Printing the values of dictionary
print(dict1.values())
```

```
dict_keys(['e', 'h', 'hey'])
dict_values([2, 4, 'why'])
```

Question Number 43

In [232…

```
# Accessing a value by its key from dictionary
print(dict1["h"])
```

```
4
```

Question Number 44

In [237…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Updating the value of an existing key
my_dict["age"] = 31

print(my_dict)
```

```
{'name': 'Alice', 'age': 31, 'city': 'New York'}
```

Question Number 45

In [240…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Removing a key-value pair using pop
removed_value = my_dict.pop("city")

print(my_dict)
print(removed_value)
```

```
{'name': 'Alice', 'age': 30}
New York
```

Question Number 46

In [251…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Get user input for the key
keys = input("Enter the key: ")
```

```
# Checking if the key exists in the dictionary
if keys not in my_dict:
    print(f"{keys} is not present in the dictionary.")
else:
    print(f"{keys} is present in the dictionary.")
```

```
age is present in the dictionary.
```

Question Number 47

In [1]:

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Getting all keys from the dictionary
keys = my_dict.keys()

# Converting the view object to a list (optional)
keys_list = list(keys)

print(keys)
print(keys_list)
```

```
dict_keys(['name', 'age', 'city'])
['name', 'age', 'city']
```

Question Number 48

In [257…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Getting all keys from the dictionary
values = my_dict.values()

# Converting the view object to a list (optional)
values_list = list(values)

print(values)
print(values_list)
```

```
dict_values(['Alice', 30, 'New York'])
['Alice', 30, 'New York']
```

Question Number 49

In [260…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Getting all key-value pairs from the dictionary
key_value_pairs = my_dict.items()

# Converting the view object to a list (optional)
key_value_pairs_list = list(key_value_pairs)

print(key_value_pairs)
print(key_value_pairs_list)
```

```
dict_items([('name', 'Alice'), ('age', 30), ('city', 'New York')])
[('name', 'Alice'), ('age', 30), ('city', 'New York')]
```

Question Number 50

In [263...

```
#Printing the length of dictionary my_dict
print(len(my_dict))
```

```
3
```

Question Number 51

In [268...

```
# List of keys
keys = ['name', 'age', 'city']

# Creating a dictionary with default values
default_value = None
my_dict = dict.fromkeys(keys, default_value)

print(my_dict)
```

```
{'name': None, 'age': None, 'city': None}
```

Question Number 52

In [271...

```
# Creating two dictionaries
dict1 = {'name': 'Alice', 'age': 30}
dict2 = {'city': 'New York', 'country': 'USA'}

# Merging the dictionaries
merged_dict = {**dict1, **dict2}

print(merged_dict)
```

```
{'name': 'Alice', 'age': 30, 'city': 'New York', 'country': 'USA'}
```

Question Number 53

In [274...

```
# Creating two lists
keys = ['name', 'age', 'city']
values = ['Alice', 30, 'New York']

# Converting the two lists into a dictionary
my_dict = dict(zip(keys, values))
```

```
print(my_dict)
```

```
{'name': 'Alice', 'age': 30, 'city': 'New York'}
```

Question Number 54

In [277…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Printing the original dictionary
print("Original dictionary:", my_dict)

# Removing all key-value pairs from the dictionary
my_dict.clear()

# Printing the dictionary after clearing
print("Dictionary after clearing:", my_dict)
```

```
Original dictionary: {'name': 'Alice', 'age': 30, 'city': 'New York'}
Dictionary after clearing: {}
```

Question Number 55

In [280…

```
# Creating a dictionary
original_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Copying the dictionary using the copy() method
copied_dict = original_dict.copy()

print("Original Dictionary:", original_dict)
print("Copied Dictionary:", copied_dict)
```

```
Original Dictionary: {'name': 'Alice', 'age': 30, 'city': 'New York'}
Copied Dictionary: {'name': 'Alice', 'age': 30, 'city': 'New York'}
```

Question Number 56

In [283…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Accessing a value with a default if the key doesn't exist
value = my_dict.get("country", "Not specified")  # Key 'country' does not exist
```

```
print("Value:", value)
```

```
Value: Not specified
```

Question Number 57

In [3]:

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Iterating over keys directly
for key in my_dict:
    print(key)
```

```
name
age
city
```

Question Number 58

In [289…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Iterating over values using the values() method
for value in my_dict.values():
    print(value)
```

```
Alice
30
New York
```

Question Number 59

In [292…

```
# Creating a dictionary
my_dict = {
    "name": "Alice",
    "age": 30,
    "city": "New York"
}

# Iterating over key-value pairs using the items() method
for key, value in my_dict.items():
    print(f"Key: {key}, Value: {value}")
```

```
Key: name, Value: Alice
Key: age, Value: 30
Key: city, Value: New York
```

Question Number 60

In [294…

```python
# Creating a list of tuples
tuple_list = [("name", "Alice"), ("age", 30), ("city", "New York")]

# Creating a dictionary from the list of tuples
my_dict = dict(tuple_list)

# Printing the resulting dictionary
print(my_dict)
```

```
{'name': 'Alice', 'age': 30, 'city': 'New York'}
```